APPENDIX
E

APPENDIX D

STANDARDS FOR THE DEFINITION PHASE

This Appendix describes the 18 possible topics to be addressed and documented in the Definition Phase of the Software Engineering Cycle.

MISSIONS AND OBJECTIVES OF USER COMPONENTS:

The missions and objectives of the user organization should be related to the computer services request. The missions of an organization may cover a range of activities, but the computer services request may relate to only one or two of the overall missions. The contents of this section will verify that the project leader has a clear understanding of the user organization's mission and that the main objectives are understood.

CURRENT SYSTEM:

The purpose of this section is to demonstrate an understanding of the present system. Include both the manual system and the computerized system if one is present. This description provides the basis for the later discussion on problems of the current system and recommended solution.

The section describes what is currently done. It should describe the inputs to the systems, the outputs, and the resources needed to convert the input to output. The resources should include both people and equipment.

The level of detail in this section will vary between projects. If the current system will remain unaffected by the new system, the description should be extensive. If the current system will be discontinued, a detailed description would probably not be profitable.

PROBLEMS WITH CURRENT SYSTEM:

A study is usually initiated because of major problems with existing systems, or new requirements. The problems with existing systems are discussed in this section. These problems should be described in sufficient detail to provide a basis for user requirements.

USER REQUIREMENTS:

The determination of user requirements is a critical step in the software engineering process. This section is the agreed upon statement which serves as the charter for developing a system design.

User requirements are precise statements of the user's needs as they relate to the user's mission. Relating user requirements to this mission is essential to the process, since without that relationship, there can be no development.

Effort must be taken to make absolutely clear what is needed. The focus must be on the functions to be supported and not on the means for implementing the functions.

User requirements can usually be grouped into four categories. They are output, input, process, and constraints. Examples follow of the type of data to be included in each:

Output Requirements:

Determine the information products that are required by the user. These should be in accordance with the mission statement. Determine the characteristics of each item as appropriate, including:

1) Performance criteria, including accuracy
2) Volume, including number and types of recipients
3) Information elements
4) Frequency of use - when needed and how often
5) Procedures for audit, distribution, and use of output
6) Security markings on the output itself

Input Requirements:

Determine the input items that are required in order to produce the products required by the user as defined in the output requirements. Determine the characteristics of each item as appropriate, including:

1) Sources of information
2) Availability of information
3) How information must be received, recorded, and handled
4) Data field definition
5) Procedures for collecting, editing, and recording data

Process Requirements:

Determine the processes that are required to produce the required output using the required input as defined above. The processes may be characterized as formula, procedures, equations, or algorithms. They are a set of predefined, exact processes which must be executed to derive the required answers in the required form from the available input.

Constraint Requirements:

Determine the specific constraints that will be imposed on the data, on the information processes and on the operation of an eventual system. Some examples are:

1) Response time for inputs and outputs
2) Reliability and availability of services needed
3) Interfaces with other systems
4) Records management requirements
5) Agency regulations

SECURITY REQUIREMENTS:

This section should include the security requirements imposed on the system. This should include physical, data, and system security. Examples are as follows:

1) Physical environmental restrictions
2) Data items, records, or file restrictions
3) System access restrictions
4) Classification of data base, output, etc.

ALTERNATIVE SOLUTIONS:

This section assesses whether a solution (manual or automated) exists for the user. For any study an answer of "it is not feasible because..." may exist. If it is determined that some user requirement or problem cannot be handled because of some explicit or implicit constraint, it is so stated here. It is possible to discuss what constraints need to be lifted or changed (and possibly, how, if immediately obvious) so as to make problem solutions possible.

If there are no restrictions apparent at this point to preclude a solution to the problems, then this section should address the investigation and alternatives for the user's request.

All sources of information used for the study should be defined in this section. For example, if a solution was pursued externally, then the sources should be indicated (i.e., GUIDE and SHARE Libraries, Federal Software Exchange, etc.) The identification of sources of information concerning resources required, limitations, etc., should also be stated.

Each viable alternative should be described in sufficient detail to provide the user with a clear understanding of the resources involved to fulfill each of the alternatives. The description for each alternative will ensure that all user requirements are fulfilled.

COSTS AND BENEFITS:

For each of the alternatives, the relationship of the cost for developing and operating the system and the benefits expected to be derived should be stated. Both one-time costs and recurring costs should be included. The benefits should include both tangible benefits for which dollar values can be assigned and non-tangible benefits. Non-tangible benefits might include improved system security or improved employee morale. The expected benefits should be supplied by the user component.

RECOMMENDATION:

Each of the alternatives presented should be addressed separately. The reasons for either rejecting or recommending an alternative should be stated.

USER RESPONSIBILITIES:

Define the functions and contributions for which the user is responsible during the development, implementation, and operation of the project. For example: design review, data entry personnel, data base manager, or other support personnel, and skills for the system should be documented.

APPLICATION GROUP'S RESPONSIBILITIES:

Define the functions for which the Application group is responsible. All deliverables should be listed in this section. This may include software, hardware, training, and documentation provided to the user.

PROPOSED SYSTEM:

This section should provide a detailed description of the system which is proposed to fulfill the user requirements. The section is divided into nine functional areas. It provides the basis for describing the development and implementation of the system.

1) Input:

Define and provide layouts of the format for any input requirements. The data source and the method for inputting data should also be defined.

2) Process:

Specify all the processes required to transfer inputs to outputs. This is typically accomplished by using flowcharts, Hierarchical Input Process Output (HIPO) diagrams, and equations.

3) Output:

Define and provide layouts of the format of all outputs. This includes hard copy reports, terminal displays, and data stored on magnetic storage media. The expected frequency and volume of the reports should also be included.

4) Data Base:

Define the data files and their fields which constitute the data base and include a description of the data fields.

5) Software:

Specify which programming languages will be used to develop the system (i.e., PL/1, FORTRAN, etc.). Any data base management systems which are to be used should be addressed (i.e., GIMS, RAMIS, NIPS, etc.).

6) Hardware:

This section describes the hardware that will be used to implement the system. This includes terminals, communication devices, computers, and any other input or output devices.

7) Conversion:

This section should describe how the data base will be loaded prior to implementation. This process may be some combination of manual support and specialized program for converting electronically stored data.

8) Installation Plan:

This section should include any changes in the user's environment which are necessary prior to cutover to the new system. This may include new manual procedures, office layout, or location for new hardware.

9) Production Responsibility:

Define the organizations which are responsible for production, maintenance, and data base backup. This includes production schedules and quality control for data base maintenance.

TEST PLAN:

The testing of a computer system is at least a two step process. First the computer specialist must be satisfied that the software functions properly. Additionally, the user must independently verify that the requirements are met.

This section should describe both types of test plans. Typically, the computer specialist and the user write their own test plans. It is important that both test plans be developed prior to the programming effort.

The test plans should match the user requirements. For example, if a requirement states that four fields are mandatory for an input transaction, then a test case should verify this aspect.

REVIEW PROCEDURES DURING DEVELOPMENT:

It is important that both the formal and informal communications exist between the computer specialist and the user during the development of the system. There are three key milestones during the design and programming phases. They are called Preliminary Design Review, Critical Design Review, and Final Design Review. A description of each follows:

1) Preliminary Design Review (PDR) - The external system requirements are reviewed for completeness. The external requirements include the first four sections listed under the Proposed System in this Appendix. They are Input, Process, Output, and Data Base. The requirements documentation should also be reviewed at this time. This would ensure that the specifications in the four sections are traceable back to user specifications.

UNCLASSIFIED

Approved For Release 2002/06/18 : CIA-RDP84-00933R000200150008-8

2) Critical Design Review (CDR) - The emphasis of this design review is on the internal specifications of the system, that is, how the system will perform its functions. System implementation (i.e., program coding, hardware procurement, etc.) normally starts after the CDR.

3) Final Design Review (FDR) - The FDR occurs between the two types of testing specified in the Test Plan section of the Proposed System in this Appendix. After the application group has completed testing, this review occurs.

In addition to specifying review procedures, describe the types of status reports, who prepares them, who receives them, and their frequency.

CHANGE CONTROL PROCEDURES DURING SOFTWARE ENGINEERING CYCLE:

The Critical Design Review forms the basis for the contract between the applications group and the user. All subsequent changes in user requirements should be handled in a formal manner. A formal change mechanism is necessary to maintain continuity with standardization of the Definition Phase. Without change control procedures, the documents describing the Design will become obsolete following the Critical Design Review.

Typically, the user requests a change and submits it to the application group where an assessment is made as to the impact on the schedule and cost of the project. This information is then available for a decision on implementing the change.

DOCUMENTATION REQUIREMENTS:

The documentation requirements should be stated. These include the documents for the life cycle maintenance of the system, user manuals, and production manuals.

SCHEDULE:

The schedule for the software engineering cycle should be stated. The milestone chart should reflect relative dates and not calendar dates. After the study has been approved, calendar dates can be substituted for the relative dates.

REVIEW PROCEDURES DURING OPERATION PHASE:

The procedures for reviewing the software after operation should be addressed in this section. It is important that both the user and application group be involved in the review. The intent is to ensure that the system continues to meet the requirements of the user.

Approved For Release 2002/06/18 : CIA-RDP84-00933R000200150008-8

UNCLASSIFIED

APPENDIX E

EXAMPLES OF TYPICAL FINISHED DOCUMENTS:

No one document is likely to contain all the 18 topics specified in APPENDIX D. The number of documents produced and the topics within each will depend on the scope and complexity of the system. Some topics (i.e., User Requirements) may be addressed in separate documents, and some topics may not be used at all for small projects.

Two examples of typical documents are specified below. They are the Feasibility Study and the Project Proposal. The Feasibility Study and Project Proposal are used for work developed within the CIA.

1) FEASIBILITY STUDY:

A Feasibility Study should be prepared when one of the following conditions exist:

- More than one viable solution is available for satisfying the requirements.

- It is not clear that the problem requires a computer solution.

- The user requirements are of such complexity that a study is required.

- The user requirements are ambiguous and a study is required to define them.

- The customer requires a rough estimate of the required resources (time, material, and cost) to determine further interest in developing a system.

At the conclusion of a Feasibility Study, the user is asked whether or not the project should be continued. The suggested topics in a Feasibility Study are:

- Mission and Objectives of User Component
- Current System
- Problems with Current System
- User Requirements
- Security Requirements
- Alternative Solutions
- Costs and Benefits of Each Solution
- Recommendation
- User Responsibilities

2) PROJECT PROPOSAL:

The Project Proposal should be prepared when one of the following conditions exist:

- It is a follow up to a Feasibility Study
- The problem is well defined
- Only one viable solution exists
- The scope of the problem can satisfactorily be addressed in a single document

The suggested topics for a Project Proposal are:

- Mission and Objectives of User Component*
- Current System*
- Problems with Current System*
- User Requirements*
- Security Requirements*
- Costs and Benefits*
- Proposed System
- Test Plan
- Application Group Resonsibilities
- User Responsibilities*
- Review Procedures During Development
- Change Control During Software Engineering Cycle
- Documentation Requirements
- Schedule
- Review Procedures During Operation Phase

*These sections are the same as in the Feasibility Study